

A Bayesian approach to type-specific conic fitting

M.J.Collett

Department of Physics, University of Auckland, New Zealand

Abstract

A perturbative approach is used to quantify the effect of noise in data points on fitted parameters in a general homogeneous linear model, and the results applied to the case of conic sections. There is an optimal choice of normalisation that minimises bias, and iteration with the correct reweighting significantly improves statistical reliability. By conditioning on an appropriate prior, an unbiased type-specific fit can be obtained. Error estimates for the conic coefficients may also be used to obtain both bias corrections and confidence intervals for other curve parameters.

1 Introduction

Linear algebraic methods for fitting conic sections to noisy data were introduced by Bookstein [1]. Subsequent work, mostly in the broad context of computer vision, has discussed topics such as iterative refinement [2], type-specific normalisation [3, 4] and the correction of obvious normalisation bias [5] or curvature bias [6, 7]. However, all of these papers confine themselves to making point estimates of the set of conic coefficients, and do not quantify the expected statistical errors.

A recent series of papers by Kanatani and coworkers [8–10] have used a perturbative treatment to predict the systematic errors introduced into the fitting process by noisy data, and hence to identify analytically (rather than merely numerically) the choices of normalisation and weighting that minimise bias and statistical error. These are forward statistical calculations, that is, the noisy data is described by a sampling distribution around the true values; the final result is an unbiased point estimate of the generic conic coefficients.

In this paper I show that we can go significantly further by also using a Bayesian (inverse) statistical treatment, that is, by considering a posterior distribution for the model parameters.

Knowing the posterior distribution has three distinct useful consequences. Firstly, and most obviously, we can place confidence intervals on the estimated fit made from any particular data set; since for the generic fit in the perturbative regime the distributions are Gaussian, equivalent results could

have been obtained from the sampling statistics. Secondly, and perhaps unexpectedly, being in possession of a distribution of fitted values rather than just a point estimate allows us to include in a systematic and unbiased fashion additional constraints, such as those required to make the fit type-specific. Finally, by propagating the estimated errors in the conic coefficients, we may obtain both confidence intervals and bias corrections for estimates of other curve parameters (e.g. the location of the centre of an ellipse).

The method used has potential application to problems other than conic fitting (e.g. camera resection [11]), so is initially developed for a generic homogeneous linear model, before being applied (with numerical examples) to the conic case. The initial sampling calculations here differ in detail from those of Kanatani *et al.*, but yield similar initial point estimates.

2 Errors in homogeneous model fitting

2.1 The model

Consider a system described by an M -component homogeneous linear model of the form

$$Z(\mathbf{x}) = \mathbf{G}^\top \mathbf{D}(\mathbf{x}) = 0 , \quad (2.1)$$

where $\mathbf{G}^\top = (g_1 \ \dots \ g_M)$ is the model vector and

$$\mathbf{D}^\top(\mathbf{x}) = (d_1(\mathbf{x}) \ \dots \ d_M(\mathbf{x})) \quad (2.2)$$

is the design vector for the Λ -dimensional data point \mathbf{x} . The elements of \mathbf{D} are known functions of \mathbf{x} (e.g. powers of the components of \mathbf{x}), and the model parameters g_m are to be fitted from the data. I confine myself here to the case of scalar Z , but a multicomponent algebraic error may be treated in similar fashion. To avoid the trivial solution $\mathbf{G} = \mathbf{0}$, we impose a normalisation constraint of the form

$$\mathbf{G}^\top \mathbf{C} \mathbf{G} = 1 . \quad (2.3)$$

The $M \times M$ constraint matrix \mathbf{C} may be constant or depend on the data; its rank R determines the dimension of the solution space.

An observed data point $\hat{\mathbf{x}}$ includes measurement error and other sources of noise,

$$\hat{\mathbf{x}} = \mathbf{x} + \Delta \mathbf{x} , \quad (2.4)$$

with the consequence that (2.1) is not exactly satisfied. Given a set of N such data points that overdetermines the solution, we therefore look for the

model vector \mathbf{G} that minimises

$$\begin{aligned} E &= \sum_{i=1}^N w_i Z^2(\hat{\mathbf{x}}_i) \\ &= \sum_i w_i \mathbf{G}^\top \mathbf{D}(\hat{\mathbf{x}}_i) \mathbf{D}^\top(\hat{\mathbf{x}}_i) \mathbf{G} = \mathbf{G}^\top \mathbf{S} \mathbf{G} , \end{aligned} \quad (2.5)$$

where

$$\mathbf{S} = \sum_{i=1}^N w_i \mathbf{D}(\hat{\mathbf{x}}_i) \mathbf{D}(\hat{\mathbf{x}}_i)^\top \quad (2.6)$$

is the $M \times M$ scatter matrix, and w_i are statistical weights to be chosen later. We still require normalisation, so that we in fact minimise the ratio

$$\frac{\mathbf{G}^\top \mathbf{S} \mathbf{G}}{\mathbf{G}^\top \mathbf{C} \mathbf{G}} , \quad (2.7)$$

which we can do by solving the generalised eigenvalue problem

$$\mathbf{S} \mathbf{G} = \lambda \mathbf{C} \mathbf{G} . \quad (2.8)$$

We choose a basis for our model such that the constraint matrix may be partitioned into an $R \times R$ upper-left corner $\tilde{\mathbf{C}}$ of full rank and zeros everywhere else,

$$\mathbf{C} = \begin{pmatrix} \tilde{\mathbf{C}} & 0_{RK} \\ 0_{KR} & 0_{KK} \end{pmatrix} , \quad (2.9)$$

where $K = M - R$. We then partition \mathbf{S} and \mathbf{G} similarly:

$$\mathbf{S} = \begin{pmatrix} \mathbf{S}_{11} & \mathbf{S}_{12} \\ \mathbf{S}_{21} & \mathbf{S}_{22} \end{pmatrix} ; \quad \mathbf{G} = \begin{pmatrix} \tilde{\mathbf{G}} \\ \tilde{\mathbf{H}} \end{pmatrix} . \quad (2.10)$$

The $K \times K$ matrix \mathbf{S}_{22} is positive definite; the $R \times R$ matrix \mathbf{S}_{11} has a zero eigenvalue in the absence of noise, but is positive definite for real data. The eigenvalue problem (2.8) is now equivalent to the reduced version

$$\tilde{\mathbf{S}} \tilde{\mathbf{G}} = \lambda \tilde{\mathbf{C}} \tilde{\mathbf{G}} , \quad (2.11)$$

where the reduced scatter matrix is the Schur complement

$$\tilde{\mathbf{S}} = \mathbf{S}_{11} - \mathbf{S}_{12} \mathbf{S}_{22}^{-1} \mathbf{S}_{21} , \quad (2.12)$$

and the full eigenvector can be reconstructed using

$$\tilde{\mathbf{H}} = -\mathbf{S}_{22}^{-1} \mathbf{S}_{21} \tilde{\mathbf{G}} . \quad (2.13)$$

$\tilde{\mathbf{S}}$ may be expressed in diagonalised form as

$$\tilde{\mathbf{S}} = \sum_{m=0}^{R-1} \tilde{\mathbf{C}} \tilde{\mathbf{G}}_m \lambda_m \tilde{\mathbf{G}}_m^\top \tilde{\mathbf{C}} , \quad (2.14)$$

and hence \mathbf{S} as

$$\mathbf{S} = \sum_m \mathbf{C} \mathbf{G}_m \lambda_m \mathbf{G}_m^\top \mathbf{C} + \begin{pmatrix} S_{12} \\ S_{22} \end{pmatrix} S_{22}^{-1} \begin{pmatrix} S_{21} & S_{22} \end{pmatrix} . \quad (2.15)$$

2.2 Perturbative treatment of errors

If there is no noise in the measured values, then the smallest eigenvalue λ_0 in (2.8) or (2.11) is zero, and the corresponding eigenvector \mathbf{G}_0 is the exact solution to the original problem. With the inclusion of noise, the eigenvector \mathbf{G}_0 will in general differ by both systematic bias and random error from the true solution.

Consider the effect on the n th eigenvalue λ_n and its associated reduced eigenvector $\tilde{\mathbf{G}}_n$ of a small change $\Delta\tilde{\mathbf{S}}$ in $\tilde{\mathbf{S}}$ (and any associated change in $\tilde{\mathbf{C}}$). (2.11) becomes

$$(\tilde{\mathbf{S}} + \Delta\tilde{\mathbf{S}}) (\tilde{\mathbf{G}}_n + \Delta\tilde{\mathbf{G}}_n) = (\lambda_n + \Delta\lambda_n) (\tilde{\mathbf{C}} + \Delta\tilde{\mathbf{C}}) (\tilde{\mathbf{G}}_n + \Delta\tilde{\mathbf{G}}_n) . \quad (2.16)$$

Expanding and keeping only first-order changes gives

$$\Delta\tilde{\mathbf{S}}\tilde{\mathbf{G}}_n + \tilde{\mathbf{S}}\Delta\tilde{\mathbf{G}}_n \simeq \Delta\lambda_n \tilde{\mathbf{C}}\tilde{\mathbf{G}}_n + \lambda_n \Delta\tilde{\mathbf{C}}\tilde{\mathbf{G}}_n + \lambda_n \tilde{\mathbf{C}}\Delta\tilde{\mathbf{G}}_n . \quad (2.17)$$

Multiplying on the left by $\tilde{\mathbf{G}}_m^\top$, and using $\tilde{\mathbf{G}}_m^\top \tilde{\mathbf{S}} = \lambda_m \tilde{\mathbf{G}}_m^\top \tilde{\mathbf{C}}$ and the orthonormality of the eigenvectors,

$$\tilde{\mathbf{G}}_m^\top \tilde{\mathbf{C}}\tilde{\mathbf{G}}_n = \delta_{mn} , \quad (2.18)$$

gives us the first-order results

$$\tilde{\mathbf{G}}_n^\top (\Delta\tilde{\mathbf{S}} - \lambda_n \Delta\tilde{\mathbf{C}}) \tilde{\mathbf{G}}_n = \Delta\lambda_n ; \quad (2.19a)$$

$$\tilde{\mathbf{G}}_m^\top (\Delta\tilde{\mathbf{S}} - \lambda_n \Delta\tilde{\mathbf{C}}) \tilde{\mathbf{G}}_n = (\lambda_n - \lambda_m) \tilde{\mathbf{G}}_m^\top \tilde{\mathbf{C}} \Delta\tilde{\mathbf{G}}_n , \quad m \neq n . \quad (2.19b)$$

Thus we have an expansion of $\Delta\tilde{\mathbf{G}}_n$ over a basis of the unperturbed eigenvectors in the form

$$\Delta\tilde{\mathbf{G}}_n = \sum_{m \neq n} \frac{\tilde{\mathbf{G}}_m \tilde{\mathbf{G}}_m^\top}{\lambda_n - \lambda_m} (\Delta\tilde{\mathbf{S}} - \lambda_n \Delta\tilde{\mathbf{C}}) \tilde{\mathbf{G}}_n . \quad (2.20)$$

Apart from the $\Delta\tilde{\mathbf{C}}$ terms (which are zero for fixed constraints and of little consequence otherwise), (2.19a) and (2.20) are precisely the results of standard first-order perturbation theory familiar from any quantum mechanics textbook.

The starting point for our perturbation is the noise-free case, for which as previously noted $\lambda_0 = 0$, and we are primarily interested in the correction to \mathbf{G}_0 , giving the simplified result that

$$\Delta\tilde{\mathbf{G}}_0 = -\tilde{\mathbf{Y}}_0 \Delta\tilde{\mathbf{S}} \tilde{\mathbf{G}}_0 , \quad (2.21)$$

where

$$\tilde{Y}_0 = \sum_{m \neq 0} \frac{\tilde{\mathbf{G}}_m \tilde{\mathbf{G}}_m^\top}{\lambda_m} \quad (2.22)$$

is a generalised inverse of the noise-free reduced scatter matrix. It is not the Moore-Penrose pseudoinverse, but is closely related to it; factorising the normalisation matrix as $\tilde{\mathbf{C}} = \mathbf{J}\mathbf{J}^\top$ we find that

$$\tilde{Y}_0 = \mathbf{J}^{-\top} \left(\mathbf{J}^{-1} \tilde{\mathbf{S}} \mathbf{J}^{-\top} \right)^+ \mathbf{J}^{-1} , \quad (2.23)$$

where superscript $+$ denotes the pseudoinverse.

To relate the change in $\tilde{\mathbf{S}}$ to underlying changes in the data points being fitted, it will be more convenient to return to the unreduced representation. To first order,

$$\Delta \tilde{\mathbf{S}} = \Delta \mathbf{S}_{11} - \mathbf{S}_{12} \mathbf{S}_{22}^{-1} \Delta \mathbf{S}_{21} - \Delta \mathbf{S}_{12} \mathbf{S}_{22}^{-1} \mathbf{S}_{21} + \mathbf{S}_{12} \mathbf{S}_{22}^{-1} \Delta \mathbf{S}_{22} \mathbf{S}_{22}^{-1} \mathbf{S}_{21} , \quad (2.24)$$

giving

$$\begin{aligned} \tilde{\mathbf{G}}_m^\top \Delta \tilde{\mathbf{S}} \tilde{\mathbf{G}}_n &= \tilde{\mathbf{G}}_m^\top \Delta \mathbf{S}_{11} \tilde{\mathbf{G}}_n + \tilde{\mathbf{H}}_m^\top \Delta \mathbf{S}_{21} \tilde{\mathbf{G}}_n + \tilde{\mathbf{G}}_m^\top \Delta \mathbf{S}_{12} \tilde{\mathbf{H}}_n + \tilde{\mathbf{H}}_m^\top \Delta \mathbf{S}_{22} \tilde{\mathbf{H}}_n \\ &= \mathbf{G}_m^\top \Delta \mathbf{S} \mathbf{G}_n ; \end{aligned} \quad (2.25)$$

a corresponding result holds trivially for $\Delta \tilde{\mathbf{C}}$. So now

$$\begin{aligned} \Delta \mathbf{G}_n &= \begin{pmatrix} \Delta \tilde{\mathbf{G}}_n \\ \Delta \tilde{\mathbf{H}}_n \end{pmatrix} = \begin{pmatrix} \Delta \tilde{\mathbf{G}}_n \\ \Delta \left(-\mathbf{S}_{22}^{-1} \mathbf{S}_{21} \tilde{\mathbf{G}}_n \right) \end{pmatrix} \\ &= \begin{pmatrix} \Delta \tilde{\mathbf{G}}_n \\ \left(-\mathbf{S}_{22}^{-1} \mathbf{S}_{21} \Delta \tilde{\mathbf{G}}_n - \mathbf{S}_{22}^{-1} \Delta \mathbf{S}_{21} \tilde{\mathbf{G}}_n \right. \\ \quad \left. + \mathbf{S}_{22}^{-1} \Delta \mathbf{S}_{22} \mathbf{S}_{22}^{-1} \mathbf{S}_{21} \tilde{\mathbf{G}}_n \right) \end{pmatrix} \\ &= \begin{pmatrix} \Delta \tilde{\mathbf{G}}_n \\ -\mathbf{S}_{22}^{-1} \mathbf{S}_{21} \Delta \tilde{\mathbf{G}}_n \end{pmatrix} - \begin{pmatrix} 0_{RR} & 0_{RK} \\ 0_{KR} & \mathbf{S}_{22}^{-1} \end{pmatrix} \Delta \mathbf{S} \begin{pmatrix} \tilde{\mathbf{G}}_n \\ -\mathbf{S}_{22}^{-1} \mathbf{S}_{21} \tilde{\mathbf{G}}_n \end{pmatrix} \\ &= -\mathbf{Y}_n (\Delta \mathbf{S} - \lambda_n \Delta \mathbf{C}) \mathbf{G}_n , \end{aligned} \quad (2.26)$$

where

$$\mathbf{Y}_n = \sum_{m \neq n} \frac{\mathbf{G}_m \mathbf{G}_m^\top}{\lambda_m - \lambda_n} + \begin{pmatrix} 0_{RR} & 0_{RK} \\ 0_{KR} & \mathbf{S}_{22}^{-1} \end{pmatrix} . \quad (2.27)$$

Again specialising to the change in \mathbf{G}_0 from the noise-free case gives

$$\Delta \mathbf{G}_0 = -\mathbf{Y}_0 \Delta \mathbf{S} \mathbf{G}_0 . \quad (2.28)$$

2.3 Dependence on measurement noise

Since the independent variables are random, a consistent calculation of the first-order change in \mathbf{S} requires us to consider not only the first-order dependence on each $\Delta \mathbf{x}_i$, but also the expected value (i.e. the deterministic part) of the second-order dependence.

The former gives

$$(\Delta \mathbf{S})_1 = \sum_{i=1}^N \sum_{\mu=1}^{\Lambda} w_i \Delta x_{i\mu} \left(\mathbf{D}_{i,\mu} \mathbf{D}_i^\top + \mathbf{D}_i \mathbf{D}_{i,\mu}^\top \right) \quad (2.29)$$

and the latter

$$(\Delta \mathbf{S})_2 = \frac{1}{2} \sum_{i\mu\nu} w_i \langle \Delta x_{i\mu} \Delta x_{i\nu} \rangle \left(\mathbf{D}_{i,\mu} \mathbf{D}_{i,\nu}^\top + \mathbf{D}_{i,\nu} \mathbf{D}_{i,\mu}^\top + \mathbf{D}_i \mathbf{D}_{i,\mu\nu}^\top + \mathbf{D}_{i,\mu\nu} \mathbf{D}_i^\top \right) \quad (2.30)$$

If the noise is homogeneous and isotropic, with $\langle \Delta x_{i\mu} \Delta x_{i\nu} \rangle = \sigma^2 \delta_{\mu\nu}$, this reduces to

$$(\Delta \mathbf{S})_2 = \sigma^2 \mathbf{C}_N + \sigma^2 (\mathbf{K} + \mathbf{K}^\top), \quad (2.31)$$

where

$$\mathbf{C}_N = \sum_{i\mu} w_i \mathbf{D}_{i,\mu} \mathbf{D}_{i,\mu}^\top; \quad (2.32a)$$

$$\mathbf{K} = \frac{1}{2} \sum_{i\mu} w_i \mathbf{D}_i \mathbf{D}_{i,\mu\mu}^\top = \frac{1}{2} \sum_i w_i \mathbf{D}_i \nabla^2 \mathbf{D}_i^\top. \quad (2.32b)$$

Substituting back into (2.19a) with $n = 0$, most terms vanish, leaving only the term in \mathbf{C}_N ,

$$\Delta \lambda_0 = \sigma^2 \mathbf{G}_0^\top \mathbf{C}_N \mathbf{G}_0; \quad (2.33)$$

if we choose our constraint matrix to be \mathbf{C}_N we have by (2.3) the simple result that $\Delta \lambda_0 = \sigma^2$.

Similarly substituting back into (2.28), some terms again vanish using the fact that the unperturbed vectors satisfy $\mathbf{D}_i^\top \mathbf{G}_0 = 0$. However, we are still left with three distinct contributions to the error in the fitted parameters:

$$\Delta \mathbf{G}_0 = -\mathbf{Y}_0 \sum_{i\mu} w_i \Delta x_{i\mu} \mathbf{D}_i \mathbf{D}_{i,\mu}^\top \mathbf{G}_0 - \sigma^2 \mathbf{Y}_0 \mathbf{C}_N \mathbf{G}_0 - \sigma^2 \mathbf{Y}_0 \mathbf{K} \mathbf{G}_0. \quad (2.34)$$

The first term is the zero-mean random error, which will be our main interest for the remainder of this paper; the second is a normalisation bias, which vanishes by (2.27) and (2.18) if we choose our constraint matrix to be \mathbf{C}_N ; the third is a curvature bias [6], which arises from the fact that $\langle Z(\hat{\mathbf{x}}_i) \rangle \neq 0$. We expect the random error to scale as σ/\sqrt{N} , and the bias terms to scale

as σ^2/R , where R is the minimum radius of curvature of the curve or surface to be fitted: for small noise ($\sigma \ll R$) and few data points the random error will dominate; for larger noise (still with $\sigma < R$) and many points the biases are more important.

To avoid the curvature bias, we need to correct (2.5) to

$$\begin{aligned} E &= \sum_i w_i (Z(\hat{\mathbf{x}}_i) - \langle Z(\hat{\mathbf{x}}_i) \rangle)^2 \\ &= \sum_i w_i \mathbf{G}^\top (\mathbf{D}(\hat{\mathbf{x}}_i) - \langle \mathbf{D}(\hat{\mathbf{x}}_i) \rangle) (\mathbf{D}(\hat{\mathbf{x}}_i) - \langle \mathbf{D}(\hat{\mathbf{x}}_i) \rangle)^\top \mathbf{G}, \end{aligned} \quad (2.35)$$

where with the assumptions already made

$$\langle \mathbf{D}(\hat{\mathbf{x}}_i) \rangle = \frac{1}{2} \sigma^2 \nabla^2 \mathbf{D}(\mathbf{x}_i) \simeq \frac{1}{2} \sigma^2 \nabla^2 \mathbf{D}(\hat{\mathbf{x}}_i). \quad (2.36)$$

If, as will commonly but not invariably be the case, the components of $\nabla^2 \mathbf{D}$ can be expressed in terms of those of \mathbf{D} as

$$\nabla^2 \mathbf{D} = 2\mathbf{L}\mathbf{D} \quad (2.37)$$

for some constant matrix \mathbf{L} , (2.35) simplifies to

$$\begin{aligned} E &= \sum_i w_i \mathbf{G}^\top (1 - \sigma^2 \mathbf{L}) \mathbf{D}(\hat{\mathbf{x}}_i) \mathbf{D}^\top(\hat{\mathbf{x}}_i) (1 - \sigma^2 \mathbf{L}^\top) \mathbf{G} \\ &= \mathbf{G}^\top (1 - \sigma^2 \mathbf{L}) \mathbf{S} (1 - \sigma^2 \mathbf{L}^\top) \mathbf{G}. \end{aligned} \quad (2.38)$$

This in turn implies that the generalised eigenvector of \mathbf{S} denoted \mathbf{G}_0 is not in fact an estimator of \mathbf{G} , but rather of $(1 - \sigma^2 \mathbf{L}^\top) \mathbf{G}$. The corrected estimator of \mathbf{G} is accordingly

$$\mathbf{G} = (1 - \sigma^2 \mathbf{L}^\top)^{-1} \mathbf{G}_0 \simeq (1 + \sigma^2 \mathbf{L}^\top) \mathbf{G}_0, \quad (2.39)$$

where we can obtain a value for σ^2 with the aid of (2.33). Under the same conditions we have that $\mathbf{K} = \mathbf{S}\mathbf{L}^\top$, and hence that the curvature bias term is $-\sigma^2 \mathbf{Y}_0 \mathbf{S} \mathbf{L}^\top \mathbf{G}_0$. Combining the two, we have a corrected curvature bias

$$(\Delta \mathbf{G})_{\mathbf{K}} = \sigma^2 (\mathbf{I}_M - \mathbf{Y}_0 \mathbf{S}) \mathbf{L}^\top \mathbf{G}_0. \quad (2.40)$$

$\mathbf{I}_M - \mathbf{Y}_0 \mathbf{S}$ projects onto \mathbf{G}_0 , so any remaining effect is purely a rescaling of the entire model vector, and hence irrelevant for a homogeneous model.

Clearly \mathbf{C}_N as given by (2.32a) is to first order an optimal choice of constraint matrix: it avoids normalisation bias, and directly gives an estimate of σ^2 and hence of the required curvature bias correction. This normalisation is equivalent both to Taubin's approximate mean-square distance [12] and to the method advocated by Harker and O'Leary [13]. In those cases, however, it appears as the result of an average over optimal weights—in the former case, those best approximating geometric distance, and in the latter,

those giving a statistically ideal least-squares fit. By contrast, we have not yet considered the question of choice of weights (though we are about to do so), only systematic bias.

A plausible alternative choice of constraint matrix, equivalent to that made in the conic case by [8], is $\mathbf{C} = \mathbf{C}_N + \mathbf{K} + \mathbf{K}^\top$, which would directly eliminate both bias terms and still leave $\Delta\lambda_0 = \sigma^2$ to first order. The downside is that this will rarely be in the desired form (2.9), requiring additional computation to change to a basis in which it is. If the reduction (2.37) (and hence the simple curvature correction (2.39)) is not available, this may nevertheless be the optimal choice. However, if the reduction (2.37) does hold, then this choice gives to first order the same results as choosing $\mathbf{C} = \mathbf{C}_N$ and applying the curvature correction, since (2.8) becomes

$$\mathbf{S}\mathbf{G} = \lambda \left(\mathbf{C}_N + \mathbf{K} + \mathbf{K}^\top \right) \mathbf{G} = \lambda \left(\mathbf{C}_N + \mathbf{S}\mathbf{L}^\top + \mathbf{L}\mathbf{S} \right) \mathbf{G} , \quad (2.41)$$

and hence

$$(\mathbf{I} - \lambda\mathbf{L})\mathbf{S}(\mathbf{I} - \lambda\mathbf{L}^\top)\mathbf{G} = \lambda\mathbf{C}_N\mathbf{G} . \quad (2.42)$$

From the first term in (2.34), the covariance matrix of the vector of coefficients is

$$\mathbf{V}_0 = \left\langle \Delta\mathbf{G}_0, \Delta\mathbf{G}_0^\top \right\rangle = \sigma^2 \mathbf{Y}_0 \sum_{i\mu} w_i^2 \mathbf{D}_i (\mathbf{D}_{i,\mu}^\top \mathbf{G}_0)^2 \mathbf{D}_i^\top \mathbf{Y}_0 . \quad (2.43)$$

By differentiating with respect to w_i we see that the optimal choice of weighting (i.e. the one that minimises the variances of the fitted parameters) is

$$w_i^{-1} = N \sum_{\mu} (\mathbf{D}_{i,\mu}^\top \mathbf{G}_0)^2 , \quad (2.44)$$

but to use this we must already have an approximate value for the model vector. It thus requires an iterative approach, in which we obtain an initial estimate for \mathbf{G}_0 using constant weights and then refine it with the improved weights. Iteration in this fashion was suggested by Sampson [2], but Sampson's weighting, based on the gradient at the measured data values, introduces bias into the fit [7] and does not converge reliably for large noise [14]. This weighting bias is similar in magnitude to the curvature bias, but its effects are more widespread, not confined to regions of high curvature. To avoid it, the gradient of the design vector $\mathbf{D}_{i,\mu}$ appearing in (2.44) must be evaluated not at the measured point $\hat{\mathbf{x}}_i$ but instead at a nearby point consistent with (the current estimate of) the model; the best method of finding this point depends on details of the model being fitted. (Note that this is a less significant issue when calculating the constraint matrix \mathbf{C}_N from (2.32a); in that case we can to leading order safely evaluate $\mathbf{D}_{i,\mu}$ at the measured points, since the summation over i averages over individual

deviations.) With the optimal weighting (2.44) the quantity E that the fit minimises is σ^{-2} times the conventional χ^2 statistic for $Z(\hat{\mathbf{x}})$; to leading order in the size of the measurement noise, this is also equal to the geometric mean-square error. Although it is important that the reweighting process not introduce bias, it does not otherwise need to be very precise; the first reweighting may give a significant reduction in the random error, but there is typically little further gain from repeated iterations. It should be stressed (since it has not always been clear in the existing literature) that the purpose of reweighting is to improve the precision of the fit, not its accuracy; it is not an effective technique for the reduction or removal of bias.

With the weighting (2.44) (or an adequate approximation to it), the resulting covariance matrix is

$$\mathbf{V}_0 = \frac{\sigma^2}{N} \mathbf{Y}_0 \sum_i w_i \mathbf{D}_i \mathbf{D}_i^\top \mathbf{Y}_0 = \frac{\sigma^2}{N} \mathbf{Y}_0 \mathbf{S} \mathbf{Y}_0 = \frac{\sigma^2}{N} \mathbf{Y}_0 ; \quad (2.45)$$

for $K = 1$ we can equivalently write

$$\Delta \mathbf{G}_0 = \eta_0 \frac{\sigma}{\sqrt{N S_{22}}} \mathbf{H}_0 + \sum_{m \neq 0} \eta_m \frac{\sigma}{\sqrt{N \lambda_m}} \mathbf{G}_m , \quad (2.46)$$

where η_m are independent unit-variance random variables and $\mathbf{H}_0^\top = (\mathbf{0}_R^\top \ 1)$. Alternatively, with a Gaussian model for the errors, the sampling distribution for $\Delta \mathbf{G}_0$ (or equivalently for $\hat{\mathbf{G}}_0 = \mathbf{G}_0 + \Delta \mathbf{G}_0$) is

$$\begin{aligned} P(\Delta \mathbf{G}_0 | \mathbf{G}_0, \sigma, \{\mathbf{x}_i\}) &\propto \exp \left(-\frac{\Delta \mathbf{G}_0^\top \mathbf{S} \Delta \mathbf{G}_0}{2\sigma^2/N} \right) \delta(\mathbf{G}_0^\top \mathbf{C} \Delta \mathbf{G}_0) \\ &\propto \exp \left(-\frac{N}{2\sigma^2} \hat{\mathbf{G}}_0^\top \mathbf{S} \hat{\mathbf{G}}_0 \right) \delta(\hat{\mathbf{G}}_0^\top \mathbf{C} \hat{\mathbf{G}}_0 - 1) , \end{aligned} \quad (2.47)$$

where the delta-function enforces normalisation, ensuring that the resulting covariance matrix is proportional to the pseudoinverse of \mathbf{S} , not the (divergent) full inverse. The individual point errors $\Delta \mathbf{x}_i$ are not present in (2.47) other than via their contributions to the scatter matrix, which is a sufficient statistic for this problem. The curvature bias correction has not been explicitly represented: $\hat{\mathbf{G}}_0$ here is the *uncorrected* vector of coefficients.

The results to this point are in practice very similar to those of [10], although there are two significant differences in approach. The first difference is that in this paper I have minimised the bias from each source separately: the normalisation matrix is chosen to remove normalisation bias; reweighting bias is minimised by evaluating the gradient at a point consistent with the current best-fit model; and the curvature bias is corrected as a separate, final step. In contrast, the method of [10] chooses the normalisation matrix to minimise all three biases simultaneously, including that from Sampson reweighting. One advantage of separate treatment is that the normalisation matrix may be calculated by the same method both before and after

reweighting. The second difference is that I have taken my perturbation expansion only to the first nonvanishing order, which in some cases is first order and in others second. The consistently second-order treatment of [8] finds an additional normalisation bias term, but this is smaller than the leading term by a factor of N and hence in the perturbative regime is always smaller than the the random error and not statistically significant.

2.4 Posterior probabilities

Although sampling distributions are useful for the general comparison of fitting methods, what we really want for an individual set of real measurements is not the distribution (2.47), which presumes knowledge of the true values that we do not have, but the posterior probability $P(\mathbf{G}|\{\hat{\mathbf{x}}_i\})$. For the Gaussian case the posterior probability and the sampling distribution are to first order interchangeable for a sufficiently broad prior, leading us to estimate the former as

$$P(\mathbf{G}|\{\hat{\mathbf{x}}_i\}\sigma) \sim \exp\left(-\frac{N}{2\sigma^2}\mathbf{G}^\top\hat{\mathbf{S}}\mathbf{G}\right)\delta(\mathbf{G}^\top\hat{\mathbf{C}}\mathbf{G}-1) . \quad (2.48)$$

In fact, noting that the optimally weighted error statistic is equivalent to the geometric mean-square error, we can obtain (2.48) directly from Bayes' theorem. With a Gaussian prior for the measurement noise we have

$$\begin{aligned} P(\hat{\mathbf{x}}|\mathbf{G}_0\sigma) &= \int d\mathbf{x} P(\hat{\mathbf{x}}|\mathbf{x}\sigma)P(\mathbf{x}|\mathbf{G}_0) \\ &= \frac{1}{(\sqrt{2\pi}\sigma)^\Lambda} \int d\mathbf{x} \exp\left(-\frac{1}{2\sigma^2}|\hat{\mathbf{x}}-\mathbf{x}|^2\right) P(\mathbf{x}|\mathbf{G}_0) . \end{aligned} \quad (2.49)$$

(2.50)

The prior probability for the (unknown) true point \mathbf{x} may be factored into a delta-function ensuring the point is on the surface $Z(\mathbf{x}) = 0$ and the distribution \tilde{P} of observed points over the surface,

$$P(\mathbf{x}|\mathbf{G}_0) = |\nabla Z(\mathbf{x})|\delta(Z(\mathbf{x}))\tilde{P}(\mathbf{x}|\mathbf{G}_0) . \quad (2.51)$$

Now write the true point in terms of the point $\bar{\mathbf{x}}$ nearest to the measured $\hat{\mathbf{x}}$ that satisfies the model and an offset $\boldsymbol{\xi}$,

$$\mathbf{x} = \bar{\mathbf{x}} + \boldsymbol{\xi} ; \quad (2.52)$$

for \mathbf{x} to be on the surface, we must have (to second order in $\boldsymbol{\xi}$)

$$Z(\mathbf{x}) \simeq Z(\bar{\mathbf{x}}) + \boldsymbol{\xi} \cdot \nabla Z(\bar{\mathbf{x}}) + \frac{1}{2} (\boldsymbol{\xi} \cdot \nabla)^2 Z(\bar{\mathbf{x}}) = 0 . \quad (2.53)$$

Since $\hat{\mathbf{x}} - \bar{\mathbf{x}}$ is parallel to $\nabla Z(\bar{\mathbf{x}})$, we can expand

$$\begin{aligned}
|\hat{\mathbf{x}} - \mathbf{x}|^2 &= |\hat{\mathbf{x}} - \bar{\mathbf{x}} - \boldsymbol{\xi}|^2 \\
&= |\hat{\mathbf{x}} - \bar{\mathbf{x}}|^2 - 2(\hat{\mathbf{x}} - \bar{\mathbf{x}}) \cdot \boldsymbol{\xi} + |\boldsymbol{\xi}|^2 \\
&= |\hat{\mathbf{x}} - \bar{\mathbf{x}}|^2 - 2 \frac{(\hat{\mathbf{x}} - \bar{\mathbf{x}}) \cdot \nabla Z(\bar{\mathbf{x}})}{|\nabla Z(\bar{\mathbf{x}})|^2} \nabla Z(\bar{\mathbf{x}}) \cdot \boldsymbol{\xi} + |\boldsymbol{\xi}|^2 \\
&= |\hat{\mathbf{x}} - \bar{\mathbf{x}}|^2 + \frac{(\hat{\mathbf{x}} - \bar{\mathbf{x}}) \cdot \nabla Z(\bar{\mathbf{x}})}{|\nabla Z(\bar{\mathbf{x}})|^2} (\boldsymbol{\xi} \cdot \nabla)^2 Z(\bar{\mathbf{x}}) + |\boldsymbol{\xi}|^2 \\
&= |\hat{\mathbf{x}} - \bar{\mathbf{x}}|^2 + \boldsymbol{\xi}^\top \mathbf{B} \boldsymbol{\xi} ,
\end{aligned} \tag{2.54}$$

where the elements of the matrix \mathbf{B} are

$$B_{\alpha\beta} = \delta_{\alpha\beta} + \frac{(\hat{\mathbf{x}} - \bar{\mathbf{x}}) \cdot \nabla Z(\bar{\mathbf{x}})}{|\nabla Z(\bar{\mathbf{x}})|^2} \partial_\alpha \partial_\beta Z(\bar{\mathbf{x}}) . \tag{2.55}$$

Substituting back into (2.50), and taking \tilde{P} to be slowly varying on the typical scale of $\Delta \mathbf{x}$, we have

$$\begin{aligned}
P(\hat{\mathbf{x}}|\mathbf{G}_0\sigma) &\simeq \frac{1}{(\sqrt{2\pi}\sigma)^\Lambda} \exp\left(-\frac{1}{2\sigma^2}|\hat{\mathbf{x}} - \bar{\mathbf{x}}|^2\right) \tilde{P}(\bar{\mathbf{x}}|\mathbf{G}_0) \\
&\quad \times \int d\boldsymbol{\xi} \exp\left(-\frac{1}{2}\boldsymbol{\xi}^\top \mathbf{B} \boldsymbol{\xi}\right) |\nabla Z(\bar{\mathbf{x}} + \boldsymbol{\xi})| \delta(Z(\bar{\mathbf{x}} + \boldsymbol{\xi})) \\
&= \frac{1}{\sqrt{2\pi\sigma^2|\mathbf{B}|}} \exp\left(-\frac{1}{2\sigma^2}|\hat{\mathbf{x}} - \bar{\mathbf{x}}|^2\right) \tilde{P}(\bar{\mathbf{x}}|\mathbf{G}_0) .
\end{aligned} \tag{2.56}$$

To leading order in $\hat{\mathbf{x}} - \bar{\mathbf{x}}$,

$$\begin{aligned}
|\mathbf{B}| &\simeq 1 + \frac{(\hat{\mathbf{x}} - \bar{\mathbf{x}}) \cdot \nabla Z(\bar{\mathbf{x}})}{|\nabla Z(\bar{\mathbf{x}})|^2} \nabla^2 Z(\bar{\mathbf{x}}) \\
&\simeq \exp\left(\frac{(\hat{\mathbf{x}} - \bar{\mathbf{x}}) \cdot \nabla Z(\bar{\mathbf{x}})}{|\nabla Z(\bar{\mathbf{x}})|^2} \nabla^2 Z(\bar{\mathbf{x}})\right) ,
\end{aligned} \tag{2.57}$$

giving

$$P(\hat{\mathbf{x}}|\mathbf{G}_0\sigma) \simeq \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma^2} \left| \hat{\mathbf{x}} - \bar{\mathbf{x}} + \frac{\sigma^2}{2} \frac{\nabla Z(\bar{\mathbf{x}})}{|\nabla Z(\bar{\mathbf{x}})|^2} \nabla^2 Z(\bar{\mathbf{x}}) \right|^2\right) \tilde{P}(\bar{\mathbf{x}}|\mathbf{G}_0) . \tag{2.58}$$

We see that the density of measured points is higher on the concave side of the surface $Z = 0$, which is an intuitively reasonable result. It is not however directly useful. To use Bayes' theorem to interchange the roles of $\hat{\mathbf{x}}$ and \mathbf{G}_0 we must work with probabilities, not probability densities (as Section 15.7 of [15] illustrates). The relevant volume element is not $d^\Lambda \mathbf{x}$ but $d|\hat{\mathbf{x}} - \bar{\mathbf{x}}| d^{\Lambda-1} \bar{\mathbf{x}}$, or equivalently $dZ(\hat{\mathbf{x}}) d^{\Lambda-1} \bar{\mathbf{x}}$, factorised into the measured

error (either geometric or algebraic) and the position on the surface, which was already integrated out in (2.56). We find that

$$\begin{aligned}
& P(\hat{Z}|\mathbf{G}_0\sigma) d\hat{Z} \\
&= \frac{1}{|\nabla \hat{Z}|} P(\hat{\mathbf{x}}|\mathbf{G}_0\sigma) d\hat{Z} \\
&\simeq \frac{d\hat{Z}}{\sqrt{2\pi\sigma} |\nabla \bar{Z}| (1 + ((\hat{\mathbf{x}} - \bar{\mathbf{x}}) \cdot \nabla) \nabla \bar{Z})} \exp \left(-\frac{1}{2\sigma^2} \frac{|\hat{Z} + \frac{\sigma^2}{2} \nabla^2 \bar{Z}|^2}{|\nabla \bar{Z}|^2} \right) \\
&\simeq \frac{d\hat{Z}}{\sqrt{2\pi\sigma} |\nabla \bar{Z}|} \exp \left(-\frac{1}{2\sigma^2} \frac{|\hat{Z} - \frac{\sigma^2}{2} \nabla^2 \bar{Z}|^2}{|\nabla \bar{Z}|^2} \right), \tag{2.59}
\end{aligned}$$

where we have expanded the gradient and again used the fact that $\hat{\mathbf{x}} - \bar{\mathbf{x}}$ is parallel to $\nabla Z(\bar{\mathbf{x}})$, more specifically that

$$\hat{\mathbf{x}} - \bar{\mathbf{x}} \simeq \frac{\nabla \bar{Z}}{|\nabla \bar{Z}|^2} \hat{Z}. \tag{2.60}$$

Thus for probabilities the result of (2.58) is reversed: for any given small area of the surface, there is a greater probability of finding a nearby point on the convex side than on the concave. This is illustrated for the 2D case in Figure 1. Taking the product over all sample points and using Bayes' theorem to convert $P(\{\hat{Z}_i\}|\mathbf{G}_0\sigma)$ to $P(\mathbf{G}|\{\hat{Z}_i\}\sigma)$ now directly yields (2.48) (including an explicit representation of the curvature correction).

While the perturbative results obtained earlier show that the method recommended in this paper is optimal among the class of linear algebraic methods, the direct derivation demonstrates the stronger result that it is globally optimal: within the small-noise approximation, no method gives a superior predictor of \mathbf{G} (though other methods may be equally good).

Since we do not know in advance the value of σ , we integrate over the Jeffreys prior [16, 15] $P(\sigma) \propto 1/\sigma$ to obtain a t -distribution,

$$P(\mathbf{G}|\{\hat{\mathbf{x}}_i\}) \sim \left(\mathbf{G}^\top \hat{\mathbf{S}} \mathbf{G} \right)^{-N/2} \delta(\mathbf{G}^\top \hat{\mathbf{C}} \mathbf{G} - 1); \tag{2.61}$$

for large N this reduces back to

$$P(\mathbf{G}|\{\hat{\mathbf{x}}_i\}) \sim \exp \left(-\frac{N}{2\hat{\sigma}^2} \mathbf{G}^\top \hat{\mathbf{S}} \mathbf{G} \right) \delta(\mathbf{G}^\top \hat{\mathbf{C}} \mathbf{G} - 1), \tag{2.62}$$

where $\hat{\sigma}^2 = \hat{\mathbf{G}}_0^\top \hat{\mathbf{S}} \hat{\mathbf{G}}_0$. The corresponding covariance matrix can be found from (2.43) with the obvious substitutions of measured for true values.

These results hold for the case of a broad prior distribution for \mathbf{G} , when the posterior distribution is just the normalised likelihood. If we wish to

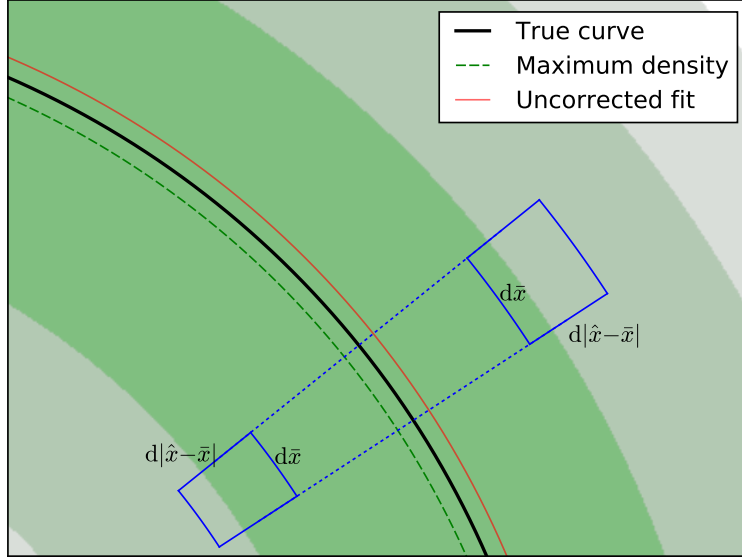


Figure 1: The probability density $P(\hat{\mathbf{x}})$ (indicated by the shading) is larger on the concave side of the curve, but for each element of the curve $d\bar{x}$ the corresponding area $dA = d\bar{x} d|\hat{\mathbf{x}} - \bar{\mathbf{x}}|$ is larger on the convex side, and so is the element of probability $P(\hat{\mathbf{x}}) dA$. Thus the uncorrected fit lies outside the true curve.

impose additional constraints on the solution, we can introduce a suitable restrictive prior and multiply it by the likelihood (2.62) to obtain the corresponding constrained posterior probability.

3 Application to conic sections

For conic fitting we have 2-dimensional data points $\mathbf{x}^\top = (x \ y)$ (in Cartesian coordinates), with the 6-component design vector

$$\mathbf{D}^\top(\mathbf{x}) = (x^2 \ xy \ y^2 \ x \ y \ 1) . \quad (3.1)$$

The self-normalising constraint matrix \mathbf{C}_N is of rank 5, and is already in the desired form (2.9); it can be constructed directly from the elements of \mathbf{S} using eqn(75) of [7]. The constant rank-3 constraint matrices preferred in earlier treatments [1, 3] inevitably result in biased estimates, and will not be considered further in this paper; the biases are exhibited numerically in [7] (note that the results therein include some for highly eccentric ellipses that are beyond the regime in which the perturbation theory used in this paper is accurate).

The curvature bias can indeed be expressed in the form (2.37), with \mathbf{L} having nonzero elements only in its last row, which is $(1 \ 0 \ 1 \ 0 \ 0 \ 0)$. From this it follows that $\mathbf{L}\mathbf{C} = \mathbf{C}\mathbf{L}^\top = 0$, so the curvature bias correction does not alter the normalisation and the residual curvature bias (2.40) vanishes completely. Furthermore, the approximate equality in (2.39) is exact, so actually performing the bias correction is simply a matter of modifying the final component g_{n6} of each eigenvector \mathbf{G}_n to

$$g'_{n6} = g_{n6} + \hat{\sigma}^2(g_{n1} + g_{n3}) , \quad (3.2)$$

where $\hat{\sigma}^2 = \lambda_0$.

For iterative reweighting according to (2.44) we may evaluate the gradient $\mathbf{D}_{i,\mu}$ at a point on the estimated curve determined with the aid of the appropriate elliptical coordinate system [7] (or confocal parabolic coordinates if the curve is a parabola, but this is vanishingly likely to happen for real data unless we force it). Introduce the coordinates (η, θ) such that

$$\mathbf{x} = \cos \theta \cosh \eta \mathbf{f}_\parallel + \sin \theta \sinh \eta \mathbf{f}_\perp + \mathbf{c} , \quad (3.3)$$

where the focal points are at $\mathbf{c} \pm \mathbf{f}_\parallel$, with $\mathbf{f}_\perp = (-f_{\parallel y} \ f_{\parallel x})^\top$; the current estimated fit is a curve of constant η if it is an ellipse or of constant θ if it is a hyperbola. Each measured point $\hat{\mathbf{x}}_i$ can be expressed in these coordinates as $(\hat{\eta}_i, \hat{\theta}_i)$; the corresponding best estimate $\bar{\mathbf{x}}_i$ is at $(\eta, \hat{\theta}_i)$ for the elliptical case or $(\hat{\eta}_i, \theta)$ for the hyperbolic.

3.1 Numerical comparisons

Figures 2 and 3 compare the sampling distribution predicted by (2.43) with a number of individual fits to randomly generated data. The original curve is an ellipse with semimajor axis 1.0 and semiminor axis 0.1, sampled at 20 points distributed along one quadrant, each sample point having a random error of standard deviation 0.001 in each direction. The fitting uses the self-normalising method (i.e. $\mathbf{C} = \mathbf{C}_N$) with the simple curvature correction (3.2). The shaded confidence intervals are bordered by contours of

$$\frac{Z(\mathbf{x})}{\sqrt{\text{Var}(Z(\mathbf{x}))}} = \frac{\mathbf{G}^\top \mathbf{D}(\mathbf{x})}{\sqrt{\mathbf{D}^\top(\mathbf{x}) \mathbf{V}_0 \mathbf{D}(\mathbf{x})}} , \quad (3.4)$$

where \mathbf{G} is the vector of coefficients corresponding to the exact curve. In the upper subfigure both the fits and the intervals are calculated using unweighted (i.e. equally weighted) data; in the lower subfigure, the fits are iterated with reweighting, and the intervals are calculated from the ideal weighting, using (2.45). There are 50 individual sample sets, so we expect that at any given point around the ellipse, typically 2 or 3 will lie outside the 2σ confidence interval, and about 14 between 1σ and 2σ ; the observed results

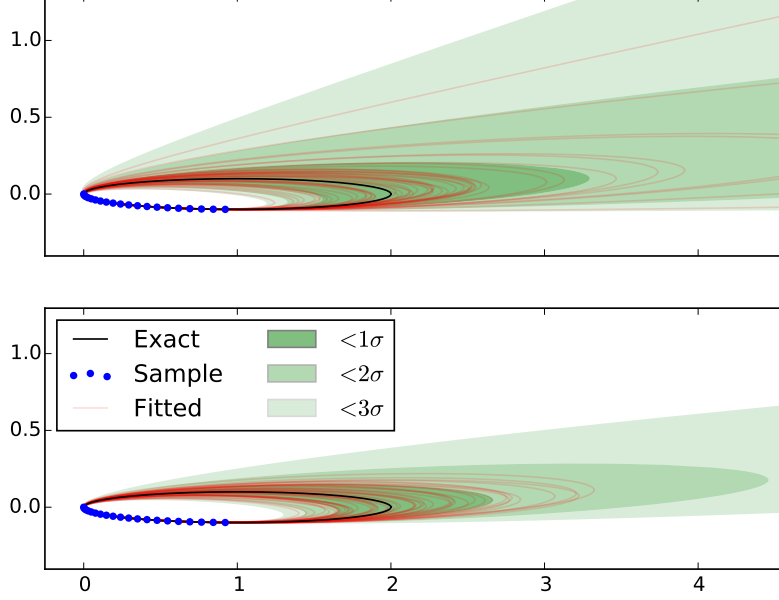


Figure 2: Comparison of predicted and observed variability in fitting. An ellipse with semimajor axis 1.0 and semiminor axis 0.1 is sampled 50 times, each sample containing 20 points distributed along one quadrant with individual measurement error 0.001. The upper subfigure is for unweighted data and the lower for weighted.

are consistent with this expectation, for both unweighted and weighted fits. We see that both predicted and observed variability is reduced by about a factor of 2 by the reweighting; it should be stressed that this does not mean that any individual fit is guaranteed to be improved by reweighting (it is perfectly possible to find examples that are made significantly worse), only that the weighted fit is more precise on average. For these parameters, the unweighted fit changes from elliptical to hyperbolic somewhere between 2 and 3 standard deviations outside the exact result, while the weighted fit is still elliptical at 3σ .

Figure 4 shows the best fit and confidence intervals estimated from a few individual samples using (2.43). Again the results are consistent with expectations: for the majority of samples the true curve is within the 1σ confidence interval, and for the rest it is within the 2σ interval; most but not all individual fits are improved by iteration with reweighting.

For the parameters used for Figs 2–4, the normalisation bias from a fixed normalisation leads to very obvious fitting errors, as illustrated in [7]. The effects of the curvature bias and the bias from Sampson’s reweighting are much smaller, but can be seen by increasing the number of sample points

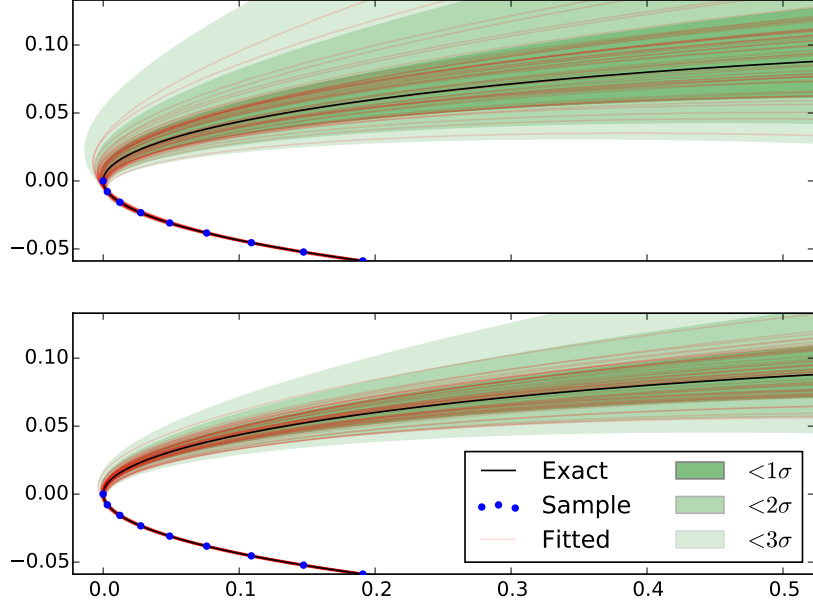


Figure 3: A close-up of Figure 2, showing the region in which the variability of the fitted curve increases as it moves away from the data points.

and the individual measurement error (still within the perturbative regime), as shown in Fig. 5.

Omitting the curvature bias correction results in fits falling outside the true curve near the tip (where the curvature is largest); reweighting using the gradient at the measured points instead of on the estimated curve results in fits that lie inside the true curve over a rather larger region. A geometric fit using a computationally expensive nonlinear least-squares method would produce results very similar to those obtained by omitting the curvature bias correction. The ‘hyper-renormalised’ method of [10] uses the Sampson weighting but modifies the normalisation matrix to correct for this: as Fig. 5 shows, this reduces the weighting bias but does not completely eliminate it.

4 Type-specific fitting

4.1 Parabolic constraint

To enforce a parabolic solution, we may use the likelihood as obtained previously in (2.48), but combine it with a prior probability that respects the type-specific constraint, giving the posterior probability with optimal

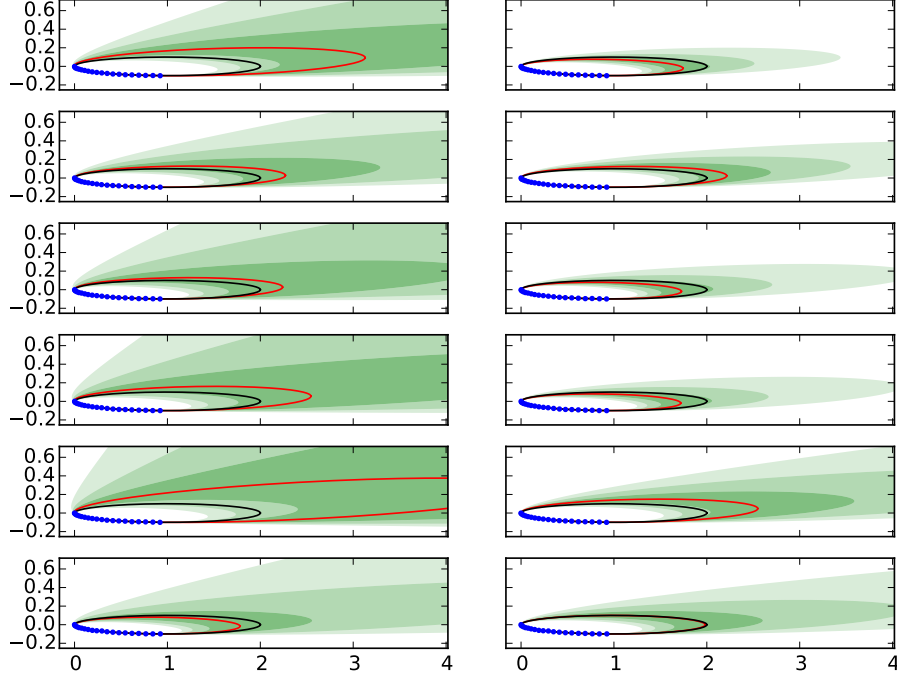


Figure 4: Posterior confidence intervals for a few of the samples from Figure 2. Unweighted fits are on the left, and corresponding weighted ones on the right. (Shading key as in Fig. 2.)

weighting as

$$\begin{aligned}
 P(\mathbf{G}|\{\hat{\mathbf{x}}_i\}) \\
 \sim \exp\left(-\frac{N}{2\hat{\sigma}^2}\mathbf{G}^\top\hat{\mathbf{S}}\mathbf{G}\right)\delta(\mathbf{G}^\top\hat{\mathbf{C}}\mathbf{G}-1)\delta(\mathbf{G}^\top\mathbf{Q}\mathbf{G}) ,
 \end{aligned}
 \tag{4.1}$$

where

$$\mathbf{G}^\top\mathbf{Q}\mathbf{G} = 4g_1g_3 - g_2^2 .
 \tag{4.2}$$

Clearly, the best estimate (both the most probable value and the posterior mean) is the value $\overline{\mathbf{G}}$ satisfying the constraint that is the shortest Mahalanobis distance from the unconstrained solution. That is, we seek the coefficient vector $\overline{\mathbf{G}}$ that minimises $\overline{\mathbf{G}}^\top\hat{\mathbf{S}}\overline{\mathbf{G}}$ subject to the constraint that $\overline{\mathbf{G}}^\top\mathbf{Q}\overline{\mathbf{G}} = 0$, while also preserving the normalisation constraint that $\overline{\mathbf{G}}^\top\hat{\mathbf{C}}\overline{\mathbf{G}} = 1$. As a first-order approximation to $\overline{\mathbf{G}}$, we have

$$\overline{\mathbf{G}} \simeq \mathbf{G}_0 - \frac{\mathbf{G}_0^\top\mathbf{Q}\mathbf{G}_0}{2\mathbf{G}_0^\top\mathbf{Q}\mathbf{Y}_0\mathbf{Q}\mathbf{G}_0}\mathbf{Y}_0\mathbf{Q}\mathbf{G}_0 ,
 \tag{4.3}$$

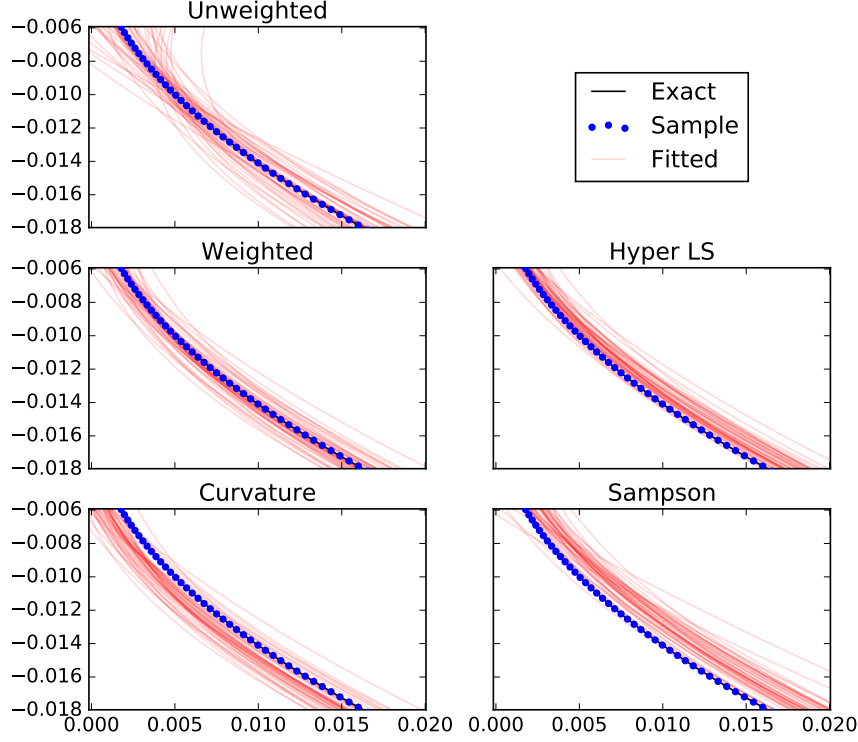


Figure 5: An ellipse with semimajor axis 1.0 and semiminor axis 0.1 is sampled 50 times, each sample containing 500 points distributed along one quadrant with individual measurement error 0.004. The top subfigure shows an unweighted fit (with curvature correction), the centre-left an unbiased weighted fit, the lower left a weighted fit without curvature bias correction and the lower right one weighted by Sampson’s gradient method. The centre-right shows the result of a ‘hyper-renormalised’ fit following the method of [10].

where \mathbf{G}_0 is the unconstrained solution. This may be iteratively refined by alternately projecting $\overline{\mathbf{G}} - \mathbf{G}_0$ onto the normal to the surface of constant $\overline{\mathbf{G}}^\top \mathbf{Q} \overline{\mathbf{G}}$ (to ensure that the length is minimised) and reapplying (4.3) with the current best estimate of $\overline{\mathbf{G}}$ replacing \mathbf{G}_0 , repeating until $\overline{\mathbf{G}}^\top \mathbf{Q} \overline{\mathbf{G}}$ is sufficiently close to zero.

Linearising the type-specific constraint around this best point estimate gives a residual Gaussian distribution of the form

$$P(\Delta \mathbf{G} | \{\hat{\mathbf{x}}_i\}) \sim \exp \left(-\frac{N}{2\hat{\sigma}^2} \Delta \mathbf{G}^\top \overline{\mathbf{S}} \Delta \mathbf{G} \right), \quad (4.4)$$

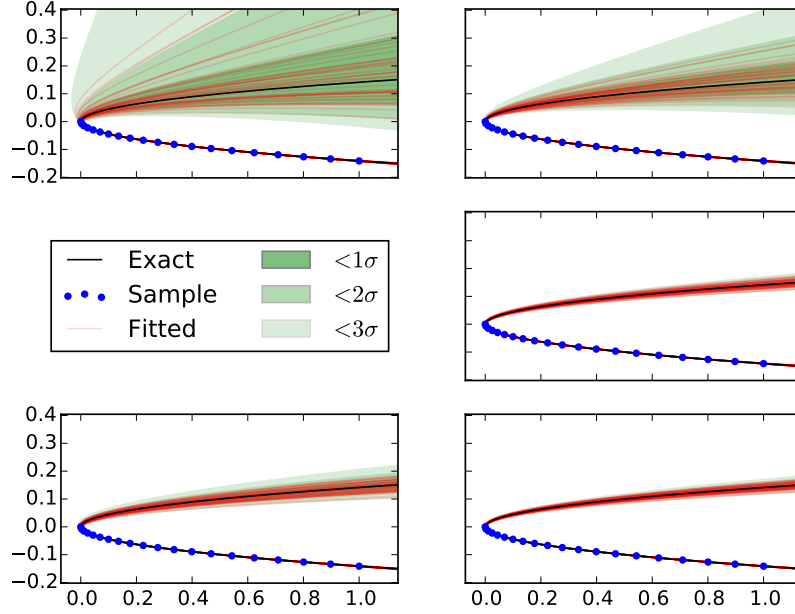


Figure 6: Comparison of variability in generic and type-specific parabolic fitting. A parabola was sampled and fitted 50 times, each sample containing 20 points with individual measurement error 0.001. The top plots are unconstrained generic fits, and the bottom ones constrained to be parabolic; the left-hand ones are unweighted and the right-hand ones optimally weighted. The centre-right plot is a hybrid case: a generic initial fit determined the weights but the weighted fit was type-specific.

where

$$\bar{S} = (1 - P_0^T - \bar{P}^T) \hat{S} (1 - P_0 - \bar{P}) ; \quad (4.5)$$

the projectors

$$P_0 = \mathbf{G}_0 \mathbf{G}_0^T \hat{\mathbf{C}} \quad \text{and} \quad \bar{P} = \frac{\hat{\mathbf{C}}^+ \mathbf{Q} \bar{\mathbf{G}} \bar{\mathbf{G}}^T \mathbf{Q}}{\bar{\mathbf{G}}^T \mathbf{Q} \hat{\mathbf{C}}^+ \mathbf{Q} \bar{\mathbf{G}}} \quad (4.6)$$

ensure that only deviations respecting both the normalisation and parabolic constraints respectively are considered. The covariance matrix is proportional to the generalised inverse \bar{Y} of the rank 4 matrix \bar{S} , calculated in the fashion specified by (2.23). We see that obtaining accurate confidence intervals is numerically more expensive for the parabolic fit than for the generic one.

Figure 6 again shows both predicted sampling distributions and a sample of individual fits to randomly generated data. The original curve is a

parabola with focal length 0.01, sampled at 20 points distributed along one arm, each sample point having a random error of standard deviation 0.001 in each direction. Comparing the lefthand plots to the righthand ones, we see as previously a moderate reduction in fitting error from reweighting. A greater improvement is apparent when comparing the type-specific fits in the bottom subfigures with the corresponding generic ones in the top subfigures. In all cases there is adequate agreement between predicted and observed variability.

The ‘hybrid’ method (generic preliminary fit and type-specific final fit) used for the centre-right subfigure gives results that are indistinguishable from the fully type-specific method at the bottom right; that is, it does not matter whether the preliminary fit used for the weighting was type-specific or not. This is not surprising: even though the initial generic and parabolic fits are very different globally, they are close together in the region of the data points, so the resulting weights are similar; and only large changes in weighting have a detectable effect on the distribution of fitted curves.

4.2 Elliptical or hyperbolic constraint

The parabolic constraint is the equality $\mathbf{G}^\top \mathbf{Q} \mathbf{G} = 0$, giving a delta-function factor in (4.1). To enforce an elliptic or hyperbolic solution requires instead the satisfaction of the corresponding *inequality*, leading to

$$P(\mathbf{G}|\{\hat{\mathbf{x}}_i\}) \sim \exp\left(-\frac{N}{2\hat{\sigma}^2} \mathbf{G}^\top \hat{\mathbf{S}} \mathbf{G}\right) \delta(\mathbf{G}^\top \hat{\mathbf{C}} \mathbf{G} - 1) u(\pm \mathbf{G}^\top \mathbf{Q} \mathbf{G}) , \quad (4.7)$$

where u is the Heaviside unit step, with the sign factor for its argument positive for the elliptic case and negative for the hyperbolic. The residual distribution is no longer locally Gaussian as (4.4) is, but is truncated along the direction normal to the surface $\mathbf{G}^\top \mathbf{Q} \mathbf{G} = 0$.

Thus the best estimate of \mathbf{G} , given by the posterior mean, is *not* the most probable value, but lies on a conic pencil passing through the unconstrained solution and the nearest point on the surface (i.e. the best parabolic fit as calculated in the previous section). Perpendicular to the pencil the distribution is still locally Gaussian, but along it we must account for the truncation. Quantitatively, we have for a one-dimensional truncated Gaussian that

$$\begin{aligned} \langle x \rangle &= \int_{x_0}^{\infty} x e^{-x^2/2} dx \Big/ \int_{x_0}^{\infty} e^{-x^2/2} dx \\ &= \sqrt{\frac{2}{\pi}} \frac{e^{-x_0^2/2}}{\text{erfc}(x_0/\sqrt{2})} . \end{aligned} \quad (4.8)$$

The estimated posterior mean for the type-constrained conic is accordingly

$$\langle \mathbf{G} \rangle = \mathbf{G}_0 + \sqrt{\frac{2}{\pi}} \frac{e^{-x_0^2/2}}{\text{erfc}(x_0/\sqrt{2})} \frac{\bar{\mathbf{G}} - \mathbf{G}_0}{x_0} , \quad (4.9)$$

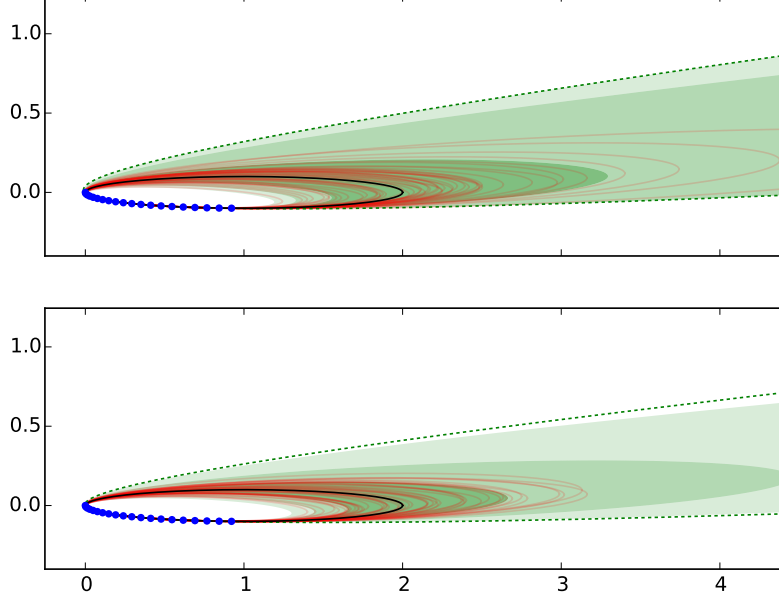


Figure 7: Type-specific ellipse fitting, using the same samples as Fig.2. As previously, the upper subfigure is for unweighted data and the lower for weighted. The dotted line is the parabola that best fits the noise-free sample points.

where \mathbf{G}_0 is the unconstrained estimate, $\overline{\mathbf{G}}$ is the nearest parabola, and $|x_0|$ is the Mahalanobis distance between the two,

$$x_0 = \pm \sqrt{N \left(\frac{\overline{\mathbf{G}}^\top \widehat{\mathbf{S}} \overline{\mathbf{G}}}{\hat{\sigma}^2} - 1 \right)}. \quad (4.10)$$

There are two distinct cases. Usually, when the unconstrained solution is already of the correct type, we take the negative sign in (4.10), and the mean will be displaced a small distance along the pencil away from the boundary. The first-order approximation (4.3) for the boundary point is sufficiently accurate. But if the unconstrained solution is of the wrong type, as may sometimes occur if the noise is large or the true solution is very close to the boundary, we need the positive sign in (4.10), and the mean will be at a point on the pencil close to the boundary, on the far side from the unconstrained solution. In this case we may require iterative refinement of the boundary point estimate, as for the parabolic constraint itself. In neither case is the mean *between* the two points defining the pencil.

It would in principle be possible to calculate the covariance matrix of an elliptical or hyperbolic solution as we have done for the generic and parabolic

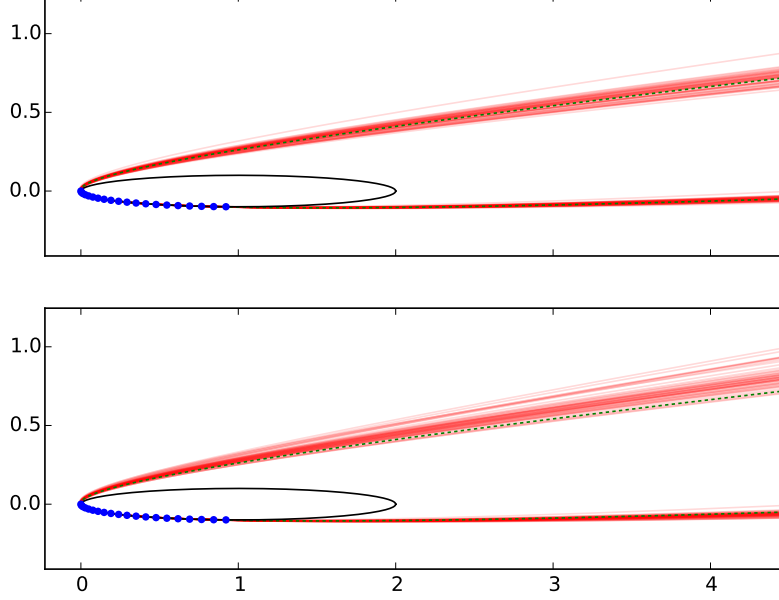


Figure 8: Forcing elliptical data to fit a parabola (upper subfigure) or a hyperbola (lower subfigure). The dotted line is again the parabola that best fits the noise-free sample points.

cases. But in this case, because the distribution is not symmetric, it is both simpler and more informative to work directly with the full parameter set of the probability distribution, namely the generic solution \mathbf{G}_0 , the generic covariance matrix \mathbf{V}_0 , and the parabolic solution $\overline{\mathbf{G}}$. The first two give the same Gaussian as the generic case, and the last the surface along which that Gaussian is truncated.

Figure 7 shows the results of ellipse-specific fits of the same samples as in Fig.2. In the unweighted case, the truncation of the sampling distribution at the parabolic solution is significant, and the resulting fits have a noticeably smaller spread than the generic ones in Fig.2. In the weighted case, for these parameters, the truncation occurs in the tail of the distribution, and the type-specific fits are accordingly little changed from the generic ones. This is appropriate behaviour for an unbiased type-specific method: if the generic fit is already clearly of the correct type, the type-specific requirement is redundant information and should not significantly alter the result. (For simplicity of presentation, the figure shows the distribution in \mathbf{x} -space truncated along the path of the parabola, but this is only an approximation to the actual truncation of the distribution in \mathbf{G} -space.)

Figure 8 shows what happens if the same elliptical data are erroneously

forced to fit a parabola or hyperbola. As is to be expected, the parabolic solutions cluster closely around, and the hyperbolic ones closely outside of, the noise-free best-fit parabola.

5 Other parameterisations

So far the discussion of fitting errors has focused on the conic coefficients \mathbf{G} ; the figures also use contours of the algebraic error Z , which is a linear function of \mathbf{G} . But there are other properties of the fitted curves that may be of interest, such as the location of the focal points, or the ratio of the axis lengths. The statistics of these may be estimated by standard error propagation techniques.

Consider a parameter ρ given by some function r of the components of \mathbf{G} . In the presence of noise, we obtain a estimated value

$$\begin{aligned}\hat{\rho} &= r(\hat{\mathbf{G}}) \\ &\simeq r(\mathbf{G}) + \sum_m \frac{\partial r}{\partial g_m} \Delta g_m + \frac{1}{2} \sum_{mn} \frac{\partial^2 r}{\partial g_m \partial g_n} \Delta g_m \Delta g_n \\ &= \rho + \Delta \mathbf{G}^\top \mathbf{r}' + \frac{1}{2} \Delta \mathbf{G}^\top \mathbf{R}'' \Delta \mathbf{G} ,\end{aligned}\tag{5.1}$$

where \mathbf{r}' and \mathbf{R}'' are the gradient and Hessian of r . Since we have ensured that $\Delta \mathbf{G}$ has zero mean, any bias in our estimate of ρ comes from the mean of the second-order term,

$$\langle \hat{\rho} - \rho \rangle = \frac{1}{2} \text{tr}(\mathbf{R}'' \mathbf{V}) ,\tag{5.2}$$

while to leading order the variance of the estimate is

$$\langle \hat{\rho}^2 \rangle - \langle \hat{\rho} \rangle^2 = \mathbf{r}'^\top \mathbf{V} \mathbf{r}' .\tag{5.3}$$

For example, we may wish to find the centre of a fitted ellipse. In the absence of noise, the centre point is

$$\mathbf{c} = - \begin{pmatrix} 2g_1 & g_2 \\ g_2 & 2g_3 \end{pmatrix}^{-1} \begin{pmatrix} g_4 \\ g_5 \end{pmatrix} .\tag{5.4}$$

With noise, the bias in our estimate is

$$\langle \Delta \mathbf{c} \rangle = \frac{1}{2} \sum_{mn} V_{mn} \frac{\partial^2 \mathbf{c}}{\partial g_m \partial g_n} ,\tag{5.5}$$

and the covariance

$$\langle \Delta \mathbf{c} \Delta \mathbf{c}^\top \rangle - \langle \Delta \mathbf{c} \rangle \langle \Delta \mathbf{c} \rangle^\top = \sum_{mn} V_{mn} \frac{\partial \mathbf{c}}{\partial g_m} \frac{\partial \mathbf{c}^\top}{\partial g_n} .\tag{5.6}$$

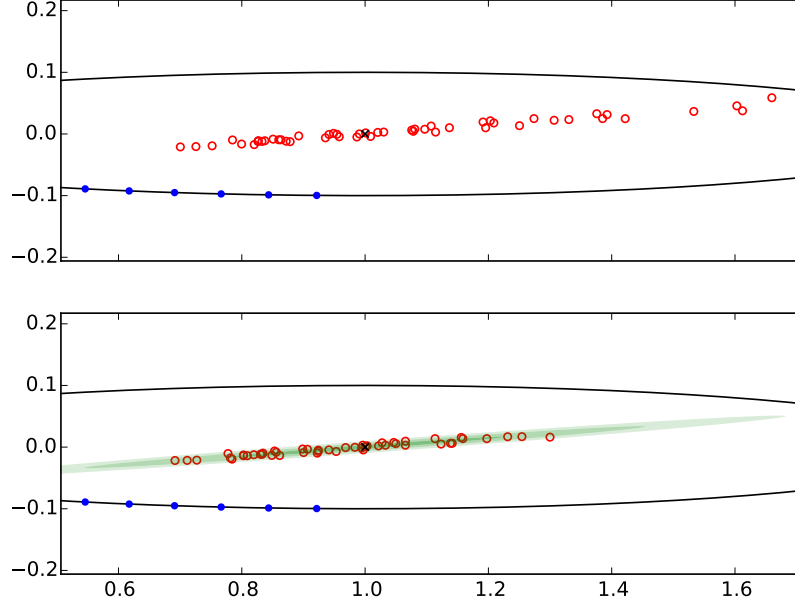


Figure 9: Top: centres of fitted ellipses, as directly calculated. Bottom: centres with bias correction, compared to the predicted sampling distribution. Sample values are open circles, and ‘x’ marks the actual ellipse centre; the predicted sampling distribution is shaded as in earlier figures.

Figure 9 illustrates the results for the same data set as used in previous figures. Without correcting for the bias, the distribution of centres is not only offset from the true value, but is noticeably skewed. Correcting each value by the corresponding expected bias (5.5) gives a distribution consistent with the spread predicted by (5.6).

6 Conclusions

The recommended algorithm for optionally type-specific conic fitting with error estimation may be summarised as follows:—

1. From the given data $\{\mathbf{x}_i\}$, construct the design vectors $\{\mathbf{D}_i\}$.
2. *Preliminary generic fit.*
 - (a) From $\{\mathbf{D}_i\}$, calculate the unweighted ($w_i = 1$) scatter matrix \mathbf{S} .
 - (b) Construct the self-normalising constraint matrix $\tilde{\mathbf{C}}_N$ and calculate the reduced scatter matrix $\tilde{\mathbf{S}}$.

- (c) Solve the generalised eigenvalue problem (2.11) to find λ_0 and \mathbf{G}_0 .
 - (d) Apply the curvature correction (3.2) to \mathbf{G}_0 .
3. *Reweighting.* Using the elliptical coordinate system (3.3) generated by \mathbf{G}_0 , for each data point \mathbf{x}_i :
- (a) find the nearest point $\bar{\mathbf{x}}_i$ on the curve $\mathbf{G}_0^\top \mathbf{D} = 0$;
 - (b) evaluate the gradient of \mathbf{D} at $\bar{\mathbf{x}}_i$; and
 - (c) calculate a new weight w_i for \mathbf{x}_i from (2.44).
4. *Weighted generic fit, with error estimate.*
- (a) Repeat step 2, but use the weights $\{w_i\}$ calculated in step 3 and find all eigenvalues and corrected eigenvectors, not only \mathbf{G}_0 .
 - (b) Construct the generalised inverse \mathbf{Y}_0 as given by (2.27). The covariance matrix for the generic fit is $\mathbf{V}_0 = \sigma^2 \mathbf{Y}_0 / N$.
5. *Type-specific fit.*
- (a) Find the nearest parabolic solution $\bar{\mathbf{G}}$ using (4.3). If the unconstrained result \mathbf{G}_0 is already of the correct type, the first-order approximation will do; otherwise iterate (4.3) as required.
 - (b) For an elliptic or hyperbolic fit, find the best point estimate of the conic parameters $\langle \mathbf{G} \rangle$ from (4.9).
 - (c) For the parabolic case, the covariance is $\bar{\mathbf{V}} = \sigma^2 \bar{\mathbf{Y}} / N$ where $\bar{\mathbf{Y}}$ is given by the generalised inverse of (4.5). For the elliptic or hyperbolic case, the probability distribution for \mathbf{G} is the same as for the generic fit (Gaussian with mean \mathbf{G}_0 , covariance \mathbf{V}_0), but truncated at the surface $\mathbf{G} = \bar{\mathbf{G}}$.

The preliminary generic fit is the same as recommended in [7], and for practical purposes equivalent to the ‘semihyper-least-squares’ method of [8].

This method contains exactly one reweighting. The effects of second or subsequent reweightings are much smaller, and are unlikely to justify the extra computation. Conversely, if no reweighting were performed, the generic covariance matrix would need to be calculated from (2.43) instead of (2.45); the extra computational effort is similar to that required to calculate the weights, with less benefit. The reweighting is based on a generic fit, even if the final fit is type-specific; we saw in Section 4.1 that making the preliminary fit type-specific has no significant effect on the result.

It has been assumed throughout that the measurement noise is sufficiently small for the leading-order perturbative treatment to be valid. Specifically, σ must be smaller than the smallest radius of curvature of the curve

being fitted. For example, the curvature correction (3.2) is not accurate for larger noise.

Provided that this assumption is satisfied, the fitting method advocated in this paper produces an unbiased, minimal variance estimator of conic coefficients. It has the advantages over previously published methods of giving confidence intervals rather than just point estimates, and of allowing type-specific fitting while remaining unbiased. It may be possible to make some of the individual steps more efficient (for example, by finding a more elegant method of extracting the mean or the covariance for the type-specific parabolic fit), but not to improve significantly on the accuracy of the result in the small-noise regime.

References

- [1] Fred L. Bookstein. Fitting conic sections to scattered data. *Computer Graphics and Image Processing*, 9(1):56–71, 1979.
- [2] P.D. Sampson. Fitting conic sections to very scattered data: An iterative refinement of the Bookstein algorithm. *Computer Graphics and Image Processing*, 18(1):97–108, 1982.
- [3] Andrew Fitzgibbon, Maurizio Pilu, and Robert B. Fisher. Direct least square fitting of ellipses. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(5):476–480, 1999.
- [4] R. Halir and J. Flusser. Numerically stable direct least squares fitting of ellipses. In *6th International Conference in Central Europe on Computer Graphics and Visualization*, pages 125–132, February 1998.
- [5] Matthew Harker, Paul O’Leary, and Paul Zsombor-Murray. Direct type-specific conic fitting and eigenvalue bias correction. *Image and Vision Computing*, 26(3):372–381, March 2008.
- [6] Kenichi Kanatani. Statistical bias of conic fitting and renormalization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(3):320–326, 1994.
- [7] M.J. Collett and G.J. Tee. Ellipse fitting for interferometry. Part 1: static methods. *Journal of the Optical Society of America A*, 31(12):2573–2583, December 2014.
- [8] Kenichi Kanatani and Prasanna Rangarajan. Hyper least squares fitting of circles and ellipses. *Computational Statistics & Data Analysis*, 55(6):2197–2208, 2011.

- [9] Kenichi Kanatani, Prasanna Rangarajan, Yasuyuki Sugaya, and Hirotaka Niitsuma. HyperLS for parameter estimation in geometric fitting. *IPSJ Transactions on Computer Vision and Applications*, 3:80–94, 2011.
- [10] Kenichi Kanatani, Ali Al-Sharadqah, Nikolai Chernov, and Yasuyuki Sugaya. Hyper-renormalization: Non-minimization approach for geometric estimation. *Information and Media Technologies*, 10(1):71–87, 2015.
- [11] M.J. Collett. Self-normalising linear camera resection. In *IVCNZ’15*, volume 30, 2015.
- [12] Gabriel Taubin. Estimation of planar curves, surfaces, and nonplanar space curves defined by implicit equations with applications to edge and range image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(11):1115–1138, 1991.
- [13] M. Harker and P. O’Leary. Direct estimation of homogeneous vectors: An ill-solved problem in computer vision. In *ICVGIP’06*, pages 919–930. Springer, 2006.
- [14] W. Gander, G.H. Golub, and R. Strebler. Least-squares fitting of circles and ellipses. *BIT Numerical Mathematics*, 34(4):558–578, 1994.
- [15] Edwin T. Jaynes. *Probability theory: the logic of science*. Cambridge University Press, 2003.
- [16] H. Jeffreys. *Theory of Probability*. The Clarendon Press, Oxford, 1939.